



Escuela  
Politécnica  
Superior

# Sistema de identificación de personas mediante uso de GAN.



Grado en Ingeniería Robótica

## Trabajo Fin de Grado

Autor:

Miguel Ángel Quevedo Mora

Tutor/es:

Miguel Cazorla

Enero 2021



Universitat d'Alacant  
Universidad de Alicante



# Sistema de identificación de personas mediante uso de GAN.

---

## **Autor**

Miguel Ángel Quevedo Mora

## **Tutor/es**

Miguel Cazorla

*Departamento de Ciencia de la Computación e Inteligencia Artificial*



Grado en Ingeniería Robótica



Escuela  
Politécnica  
Superior



Universitat d'Alacant  
Universidad de Alicante

ALICANTE, Enero 2021





# Preámbulo

En este proyecto se pretende abordar, desde un enfoque de las arquitecturas profundas, el problema de la reidentificación de personas, con objeto de realizar el seguimiento de un grupo de personas, en un entorno multicámara cerrado. El objetivo principal es realizar un control de acceso, vigilancia, inferencia de datos, o asistencia a personas mayores en residencias de ancianos.



Este trabajo no habría sido posible sin el apoyo de mi tutor Miguel Ángel Cazorla, quien me ha facilitado las herramientas necesarias para su correcto desempeño, ha tenido la paciencia y para realizar videollamadas para resolver dudas de última hora o encontrar ese error tonto en 400 y pico líneas de código. Mención especial a Fran el target de color amarillo que se ha estado moviendo por mi monitor día sí y día también.

También me gustaría hacer mención al apoyo de mis compañeros de clase y del colegio, más de uno me ha echado un cable en momentos donde ni yo mismo sabía lo que estaba pasando.

No puedo terminar sin agradecer a mi familia, que pese a nuestros roces han estado siempre ahí de buena gana aunque a veces se les escapara un

"Menudo ingeniero." "Esperad que ya llega el listo." "¿Y para eso estudias?" "A ver cuándo nos haces un robot de esos."

No podría hablar de familia sin acordarme de mi segunda familia, el Grupo Scout Brownsea, han formado parte de mi vida durante 13 años y a pesar de llegó mi momento de dejarlo atrás, siempre los llevaré en el corazón.

En último lugar dejo a mi Abuela, María Rámirez López, la santa paciencia hecha persona. Cuyas patatas fritas y albóndigas me han dado fuerzas para terminar este documento.

Y a mi Abuelo, Don José Mora Rico, del que he heredado mi cabezonería y gusto por los dulces, y para que negarlo, mi mal genio. Ojalá estuvieras aquí para reírnos como entonces, solo pido ser la mitad de hombre de lo que tú eras.

Es a ellos a quien dedico este trabajo.



*A mi abuelo Don José Mora Rico,  
dejaste un hueco que nunca podré llenar.*



*En la vida siempre perdemos nuestro rumbo.  
Pero es la gente, y no las señales,  
quienes nos guían de vuelta al camino correcto.*

Nariyoshi Miyagi.  
Cobra Kai, T3 ep4 min 29 "El camino correcto".





# Índice general

<b>1</b>	<b>Introducción:</b>	<b>1</b>
1.1	Problema a resolver. . . . .	2
1.2	Motivación. . . . .	2
1.3	Estructura del TFG. . . . .	3
<b>2</b>	<b>Marco Teórico</b>	<b>5</b>
2.1	Formas de representar la persona a reidentificar . . . . .	5
2.1.1	Representación mediante una imagen . . . . .	5
2.1.2	Representación mediante una secuencia de vídeo . . . . .	5
2.1.3	Representación mediante una descripción de texto . . . . .	6
2.2	Las dificultades y problemas en la reidentificación de personas. . . . .	7
<b>3</b>	<b>Objetivos</b>	<b>9</b>
<b>4</b>	<b>Metodología</b>	<b>11</b>
4.1	Aproximación . . . . .	11
4.2	Herramientas utilizadas. . . . .	12
4.2.1	Sistema de cámaras, hardware y descripción del entorno. . . . .	15
4.2.1.1	La cámara Reolink RLC-410W: . . . . .	16
4.2.1.2	El entorno de pruebas: . . . . .	17
4.2.2	Ordenadores utilizados en el desarrollo . . . . .	18
4.3	El dataset market-1501 . . . . .	20
4.3.1	Características generales . . . . .	20
4.3.2	Organización de los ficheros. . . . .	20
4.3.3	Características de las imágenes . . . . .	21
4.4	Instalación y entorno virtual en CONDA, prerequisites y librerías. . . . .	21
4.4.1	Instalación de dependencias Reidentificación de personas. . . . .	21
<b>5</b>	<b>Desarrollo.</b>	<b>25</b>
5.1	Pipeline del sistema de reidentificación de personas. . . . .	25
5.2	Dataset . . . . .	26
5.3	Realización de la reID . . . . .	28
5.3.1	Re-ID por imágenes del dataset . . . . .	28
5.3.2	Re-ID en tiempo real . . . . .	28
5.4	El entrenamiento de la red con Market-1501 . . . . .	29
5.4.1	Entrenar bajo la propuesta de mixed training . . . . .	29
<b>6</b>	<b>Resultados</b>	<b>33</b>
<b>7</b>	<b>Conclusiones</b>	<b>37</b>

<b>Bibliografía</b>	<b>39</b>
<b>Lista de Acrónimos y Abreviaturas</b>	<b>43</b>

# Índice de figuras

1.1	El problema de reidentificación, [2]. . . . .	1
1.2	Ejemplo de una reidentificación en entorno abierto, [34]. . . . .	2
2.1	Pipeline de una reidentificación (Re-ID) mediante una sólo imagen [1]. . . . .	6
2.2	workflow de una Re-ID mediante una secuencia. [26]. . . . .	6
2.3	workflow de una Re-ID mediante texto [30]. . . . .	7
4.1	Ejemplos de diferentes generaciones de datos sintéticos con redes generativas. . . . .	13
4.2	Diagrama cámaras y zonas. . . . .	15
4.3	Dimensiones facilitadas por el fabricante[22]. . . . .	16
4.4	Tabla de especificaciones de la cámara usada en el sistema [22]. . . . .	17
4.5	Las zonas a vista de cámara. . . . .	18
4.6	Contenido de Market-1501 individuo 0026. . . . .	20
4.7	<i>El <b>name rule</b> obedece lo siguiente:</i> <i>A: Se corresponde con la identificación del individuo dentro del dataset, de 0000 a 1501. B: Se corresponde con los datos de la cámara (de 1 a 6 en el dataset) C: Información extra no relevante para el pipeline. . . . .</i>	21
5.1	Workflow de alto nivel de nuestra aproximación a Re-ID. . . . .	25
5.2	Workflow de la segmentación del target 1502 (Fran) en la sección 1 del entorno. . . . .	26
5.3	Los diferentes vídeos generados por nuestro sistema de cámaras. . . . .	27
5.4	El proceso de cribado de la segmentación con yolo de los targets se hace a mano, eliminado aquellas imágenes no prometedoras. . . . .	28
5.5	Workflow de la reidentificación del sujeto 1503 (Miguel). . . . .	29
5.6	Workflow de la reidentificación del sujeto 1503 (Miguel) en tiempo real. . . . .	30
5.7	Visión general esquemática de la DG-Net, mixed train, con red GAN. . . . .	31
6.1	Ejecución del pipeline para un query sin imágenes del mismo en el Market-1501. . . . .	34
6.2	Oclusión parcial de un query del target 1502, con un Re-id exitoso. . . . .	34
6.3	Un query del sujeto 1503, sin imágenes del mismo en la galería. . . . .	34
6.4	Ejemplo de ejecución para un query con oclusión parcial(sólo torso superior). . . . .	34
6.5	Correcta Re-Id del sujeto 1503. . . . .	35
6.6	Correcta reidentificación del sujeto 1502 en el entorno C2, con mala iluminación. . . . .	35
6.7	Ejemplo de re-id, del target 1503. . . . .	35
6.8	Detección del sujeto 1502, con diferentes poses como resultado. . . . .	35
6.9	Detección del sujeto 1503 tomada con un dispositivo móvil, en otra fecha, con ropa similar, y distinta iluminación. . . . .	36

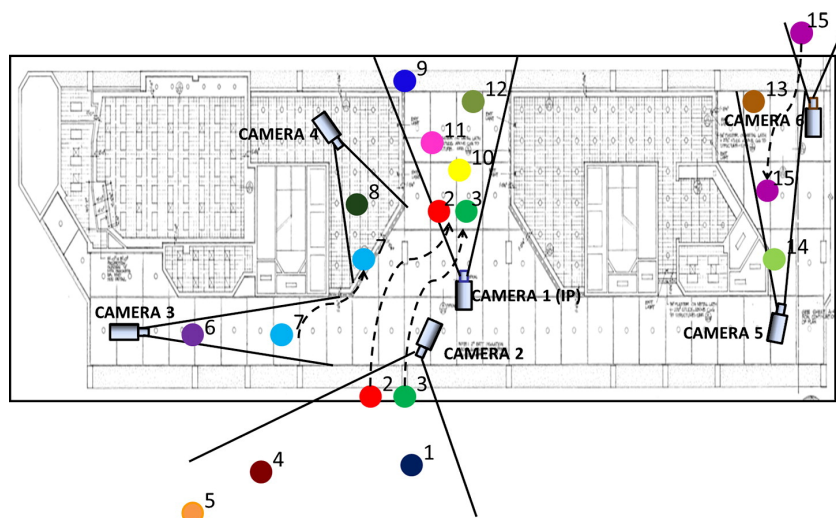


# 1 Introducción:

La reidentificación de personas es una tarea fundamental en la vigilancia automatizada por vídeo y ha sido un área de intensa investigación en los últimos años.

Dada una imagen/vídeo de una persona tomada desde una cámara, la Re-ID es el proceso de identificar a la persona a partir de imágenes/vídeos tomados desde una cámara diferente.

La Re-ID es indispensable para establecer un etiquetado coherente entre múltiples cámaras o incluso dentro de la misma cámara para restablecer un seguimiento después de un desconexión o pérdida del sistema. Además de la vigilancia, tiene aplicaciones en la robótica, la multimedia y la medicina forense. La Re-ID de personas es un problema difícil debido a la ambigüedad visual y a la incertidumbre espacio-temporal en el aspecto de una persona a través de las diferentes cámaras. Estas dificultades se ven a menudo agravadas por las imágenes de baja resolución o la mala calidad de los vídeos con grandes cantidades de información no relacionada en ellos que no ayuda a la Re-ID. Las condiciones espaciales o temporales para limitar el problema son difíciles de captar. Sin embargo, el problema ha recibido una atención considerable de la comunidad de investigadores de la visión artificial, y campos relacionados, debido a su amplia aplicabilidad y utilidad [2].



**Figura 1.1:** El problema de reidentificación, [2].

En este Trabajo Final de Grado (TFG), exploramos el problema de la Re-ID de personas y con objeto de establecer un pipeline capaz de realizar esa reidentificación en un entorno multicámara.

## 1.1 Problema a resolver.

Este TFG tiene cómo objetivo resolver el problema de la Re-ID en un entorno cerrado, con múltiples cámaras, sin oclusión en las mismas. Se plantea resolverlo desde la perspectiva del deep-learning.

Un ejemplo del problema se puede ver en la figura 1.1, Tenemos varias cámaras, las personas representadas por diferentes puntos de colores, se pueden mover libremente por el recinto. El problema es tener identificada de manera única a cada persona, independientemente de dónde se encuentren.

Hay múltiples enfoques para realizar esta Re-ID, siguiendo la filosofía del deep-learning, enfoques que se estudiarán en la sección de **Marco teórico**. Sin embargo, nosotros lo haremos desde un enfoque en el que realizamos un estudio de las características visuales del individuo completo, concretamente en su forma y ropa, es decir no empleamos reconocimiento de caras.

Partimos de un dataset de targets, nuestro pipeline tendrá que realizar un seguimiento de esos individuos en el entorno multicámara, discriminándolos entre ellos y otras personas ajenas a la Re-ID.

## 1.2 Motivación.

Como ya se ha comentado en la introducción, la reidentificación de personas se emplea de forma activa en la vigilancia automatizada, pero no sólo en el campo de la vigilancia, también se aplica en múltiples campos como la robótica o el sector médico.

Una de las motivaciones para realizar este trabajo es su posible aplicación en el seguimiento de ancianos en un residencia. Nos permite realizar un seguimiento de los residentes en todo momento, monitorizando su estado, permitiendo así actuar de manera más eficaz ante un comportamiento anómalo (caídas, inactividad prolongada etc.)

Otra aplicación interesante puede ser el control de personas en un supermercado. Con la Re-ID podríamos realizar un seguimiento del movimiento de un determinado cliente o de un grupo de los mismos, e inferir información, lo que nos puede permitir a nivel global una mejor distribución de productos, o incluso implementar un sistema de fidelización de clientes.



**Figura 1.2:** Ejemplo de una reidentificación en entorno abierto, [34].

## 1.3 Estructura del TFG.

A continuación realizamos un breve resumen del contenido del resto de capítulos del presente documento.

- **2 Marco Teórico:**

En este capítulo se van a revisar las diferentes tendencias y aproximaciones a la resolución del problema de la Re-ID de personas.

Se mencionan las diferentes formas de representar a la persona a Re-ID y se mencionarán las dificultades y problemas a los que se tiene que enfrentar un sistema de Re-ID.

- **3 Objetivos:**

Una lista de objetivos generales a cumplir durante la realización del proyecto.

- **4 Metodología:**

En este capítulo se habla de las herramientas utilizadas (sistema de cámaras, hardware y software), la descripción del entorno, Se habla del dataset Market-1501 y finalmente se dan tips para la Instalación de las librerías en un entorno virtual en CONDA.

- **5 Desarrollo:**

En este apartado se describe el Pipeline de Re-ID, desde la obtención del dataset de targets hasta la ejecución de la reidentificación. Además se menciona como se realiza un entrenamiento con la red neuronal y el dataset Market-1501.

- **6 Resultados:**

En esta sección se presentan diferentes resultados, aquellos que han tenido éxito, y aquellos que presentan fallos, y se trata de argumentar a qué se deben estos errores.

- **7 Conclusiones:**

Se presenta una revisión del trabajo realizado, los pros y contras de la aproximación elegida y cómo se podrían sortear.





## 2 Marco Teórico

En este capítulo se van a revisar las diferentes tendencias y aproximaciones a la resolución del problema de la Re-ID de personas.

La identificación de personas, ha sido ampliamente estudiada como un problema específico de recuperación de personas a través de cámaras no superpuestas [1]. Dada una persona que queremos identificar o monitorizar, el objetivo de la Re-ID es determinar si esta persona ha aparecido en otro lugar en un momento distinto capturado por una cámara diferente, o incluso la misma cámara en un instante de tiempo distinto [9].

### 2.1 Formas de representar la persona a reidentificar

La persona en cuestión puede ser representada entre otros por:

1. una imagen [1].
2. una secuencia de vídeo [26].
3. una descripción de texto [30].

#### 2.1.1 Representación mediante una imagen

Mediante un diseño basado en una arquitectura profunda y una buena estrategia de entrenamiento, este enfoque construye representaciones globales que capturan los detalles sutiles requeridos para la Re-ID de la persona. Entrenándose para detectar las áreas significativas de una imagen para responder fuertemente a las regiones discriminatorias como la mochila o el dobladillo de los pantalones cortos. En la Figura 2.1 Los mapas de calor indican regiones de la imagen que activan fuertemente las diferentes dimensiones de incrustación.

#### 2.1.2 Representación mediante una secuencia de vídeo

En este tipo de aproximación se emplean secuencias de imágenes, como se puede ver en la Figura 2.2:

- (a) Secuencias de imágenes.
- (b) Generación de grupos de fragmentos candidatos.
- (c) Creación de pares de fragmentos candidatos
- (d) Clasificación como instancias positivas y negativas.
- (e) Seleccionar y clasificar simultáneamente los pares de fragmentos más discriminantes.

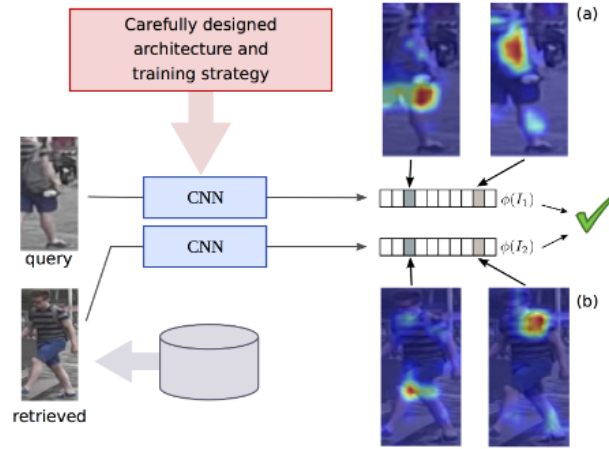


Figura 2.1: Pipeline de una Re-ID mediante una sola imagen [1].

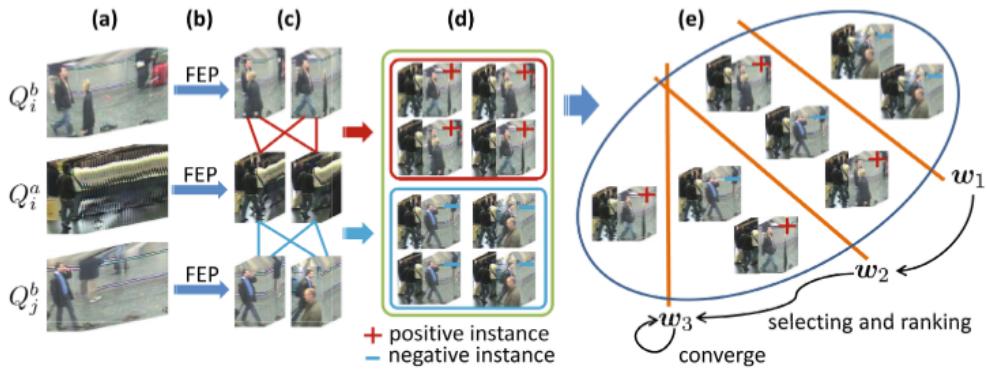


Figura 2.2: workflow de una Re-ID mediante una secuencia. [26].

### 2.1.3 Representación mediante una descripción de texto

En los casos anteriores tenemos una imagen (*caso 1*) o grupos de imágenes (*caso 2*), de la que partir para realizar la Re-ID. Sin embargo, es posible que la imagen visual inicial no esté disponible en algunas aplicaciones prácticas.

Por ejemplo, el delincuente se describe mediante una descripción de texto de manera indirecta, "Una mujer joven vestida de rojo casual con una mochila". Los métodos tradicionales no pueden abordar esta cuestión. Por ello sobre la base de un conjunto de atributos predefinidos la consulta de la descripción de texto puede transformarse en un atributo vectorial, por lo que puede utilizarse para la recuperación en el conjunto de la galería. Sin embargo, los atributos proporcionados por el usuario son a veces incompletos. Esto es un nuevo tipo de problema, que no se abordará en este TFG, dado que es una situación que no aplica en nuestra resolución del problema. Para saber más, ver la Figura 2.3.

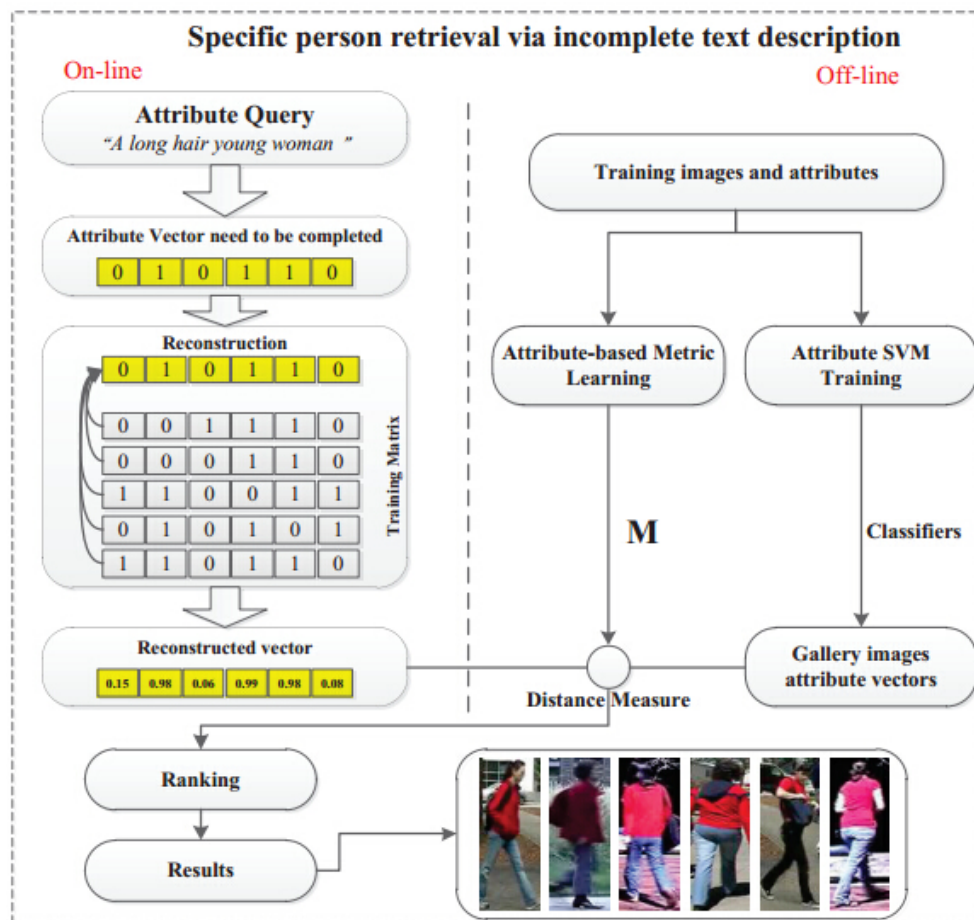


Figura 2.3: workflow de una Re-ID mediante texto [30].

## 2.2 Las dificultades y problemas en la reidentificación de personas.

El Re-ID es una tarea difícil debido a:

- La presencia de diferentes puntos de vista [16].
- Diferentes resoluciones de imagen [27].
- Cambios de iluminación [15].
- Posturas no restringidas [23].
- Oclusiones [14].
- Modalidades heterogéneas [28].
- Entornos de cámara complejos.
- Desorden de fondo [13].

- Generaciones de cajas delimitadoras (bounding boxes) poco fiables, etc.

Todo ello da lugar a variaciones e incertidumbres variables. Además existen otros problemas relacionados. Estos son algunos de ellos:

- La actualización dinámica de la red de cámaras [6].
- La galería a gran escala con una recuperación eficiente [8].
- La incertidumbre del grupo [32].
- El importante cambio de dominio [5].
- Los escenarios de prueba no vistos [24].
- La actualización incremental del modelo [4].
- El cambio de ropa [29].

Estas dificultades hacen que el problema de la Re-ID siga siendo un problema sin resolver. Los primeros esfuerzos de investigación se han centrado principalmente en la construcción artesanal de rasgos con estructuras corporales [11], o en el aprendizaje métrico a distancia [18]. Con el avance del deep-learning, la Re-ID de personas ha logrado un gran desempeño. Sin embargo, todavía existe una gran brecha entre los escenarios orientados a la investigación y las aplicaciones prácticas [19].

---

### 3 Objetivos

El objetivo principal de este proyecto es el de desarrollar un sistema de reidentificación de personas, en un entorno multi-cámara. En esencia el proyecto debe de ser capaz de discernir entre una base de sujetos, y realizar un correcto etiquetado de los mismos en un entorno de diferentes cámaras.

Para conseguir este objetivo se van a seguir las diferentes fases.

- Investigar sobre el estado del arte en reidentificación mediante sistemas con el enfoque deep learning.
- Buscar y analizar las diferentes opciones viables para un reidentificado en tiempo real.
- Encontrar y mejorar una implementación que pueda cubrir nuestro problema.
- Perfeccionar dicha implementación para su uso en tiempo real.
- Estudiar un método de preprocesado previo que acelere la reidentificación obteniendo un mejor resultado final.
- Testear el sistema en diferentes entornos y situaciones.



## 4 Metodología

El objetivo de este capítulo, es reflejar el proceso de desarrollo del proyecto. Para ello, en una primera sección se explican los pasos que se han seguido y después en una segunda sección, se exponen las herramientas que hemos utilizado.

### 4.1 Aproximación

Para conseguir nuestro objetivo de detectar y reidentificar personas en un entorno multi-cámara, nos hemos servido de una integración en dos pasos:

- Una detección mediante YOLO de las personas en la escena.
- Una red preentrenada que es capaz de calcular la distancia entre la imagen query y las del dataset (dataset que incluye las imágenes de los sujetos a reidentificar). Esta red es la encargada de diferenciar las diferentes personas en la escena.

La secuencia de pasos que se han seguido para el desarrollo del proyecto es la siguiente:

1. **Valoración de las alternativas para realizar una reidentificación, estudio sobre redes GAN y métodos de deep learning para abordar el problema.** En este punto, se han estudiado las diferentes alternativas para abordar el problema desde la filosofía del deep learning. Se ha profundizado en su funcionamiento, con objeto de determinar la técnica que mejor se ajusta a nuestro proyecto.
2. **Búsqueda de datasets y sistemas preentrenados para resolver el problema en estático.** Se ha buscado un sistema preentrenado, con opción de realizar un entrenamiento propio de ser necesario, aunque finalmente se ha optado por usar el modelo de red entrenada con el dataset market-1501. Se han explorado otros dataset, para a futuro poder reentrenar el sistema de ser necesario.
3. **Desarrollo de un módulo de segmentación de personas.** El funcionamiento de este módulo es clave, dado que buscamos segmentar las personas de un frame lo mas rápido posible, puesto que se pretende trabajar en tiempo real. Se ha optado por una segmentación mediante **YOLO**, concretamente YOLOv3 [21], ya que es una manera rápida y eficaz de realizar la segmentación de un vídeo, frame a frame de nuestro target, las personas a reidentificar.
4. **Implementación de un sistema de tratamiento de imágenes para procesar nuestro propio dataset.** Partiendo del script desarrollado en el punto anterior, creamos un módulo capaz de tratar un vídeo en el que sólo sale uno de nuestros targets,

etiquetarlo correctamente y guardarlo en una carpeta con el tamaño adecuado, para que sea identificado por nuestra red en pytorch. Es decir, imágenes de 128 por 68 píxeles. Con este script podremos generar nuestro propio dataset de los **targets** (personas a reidentificar). Para realizar nuestro dataset de targets, se han grabado vídeos en las diferentes salas de nuestro sistema de cámaras objetivo, de cada target por separado. A estos vídeos, se les ha aplicado el script para obtener imágenes correctamente etiquetadas, que nuestra red pueda usar para realizar la reidentificación en un entorno multi-target.

5. **Sistema de reidentificación de personas.** Se ha optado por una red preentrenada en pytorch, en el entrenamiento se realiza un entrenamiento convencional con el dataset market (ver Sección 4.3) y redes GAN, de manera simultánea.

Esta red es capaz de calcular las distancias de una imagen query, sobre una galería, devolviendo un vector de distancias ordenado de menor a mayor. Los primeros candidatos de ese vector son las imágenes que menos distancia presentan con nuestro query, y por ello son candidatas a ser la misma persona, captada en otro momento por esa cámara u otra cámara del sistema.

6. **Adecuación del método estático a un método en tiempo real.** Esta es la última parte del proyecto, aquí se reúnen todos los módulos y scripts anteriormente desarrollados.

Con objeto de pasar el sistema a un funcionamiento en tiempo real, yolo nos facilita la segmentación rápida y robusta de las personas en una captura de vídeo, la red de pytorch realiza la carga de la galería de targets, calcula las distancias respecto a la imagen target (obtenida en tiempo real mediante yolo) y realiza la predicción. Todo esto se procesa y se muestra en el vídeo con un bounding box y el nombre del target reidentificado.

7. **Experimentación.**

## 4.2 Herramientas utilizadas.

En esta sección se van a explicar las herramientas que se han utilizado para el desarrollo del proyecto, comenzando por una descripción de las tecnologías utilizadas e incluyendo, a continuación, una subsección en el que se describen las especificaciones de los ordenadores que se han utilizado así como del entorno multicámara empleado para la experimentación.

1. **YOLO "You Only Look Once".**

Este bloque es clave, es el bloque que nos ha permitido realizar el dataset de los sujetos a reidentificar, además de permitirnos segmentar cada frame de un vídeo en tiempo real, para realizar la posterior Re-ID.

El algoritmo es un sistema de código abierto del estado del arte para detección de objetos en tiempo real, el cual hace uso de una única red neuronal convolucional para detectar objetos en imágenes. Para su funcionamiento, la red neuronal divide la imagen en regiones, prediciendo cuadros de identificación y probabilidades por cada región; las

---



cajas son ponderadas a partir de las probabilidades predichas. El algoritmo aprende representaciones generalizables de los objetos, permitiendo un bajo error de detección para entradas nuevas, diferentes al conjunto de datos de entrenamiento.

2. **Redes Generativas Antagónicas (GAN).** Una *Generative Adversarial Networks* (GAN) es una arquitectura de red neuronal diseñada para resolver el problema del modelado generativo. El objetivo de un modelo generativo es estudiar una colección de ejemplos de capacitación y conocer la distribución de probabilidad que los generó. Las GAN pueden entonces generar más ejemplos a partir de la distribución de probabilidad estimada. Los modelos generativos basados en el aprendizaje profundo son comunes.

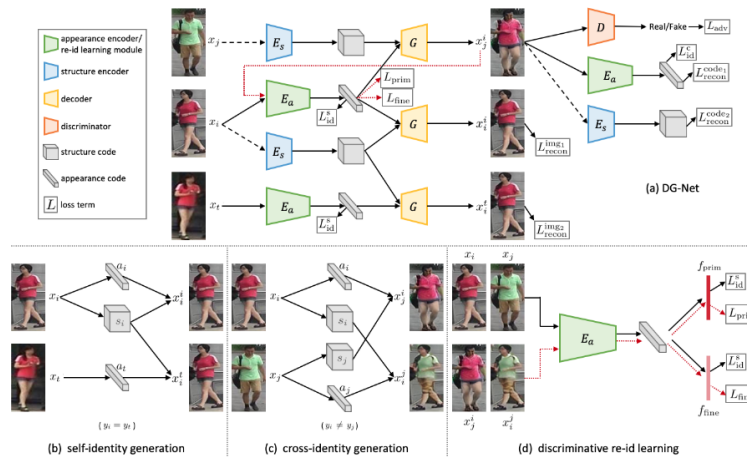
Las GAN se han aplicado con éxito a una amplia variedad de tareas (sobre todo en entornos de investigación), pero siguen presentando desafíos y oportunidades de investigación únicos porque se basan en la teoría de los juegos, mientras que la mayoría de los demás enfoques de la modelización generativa se basan en la optimización [10].

Estas redes nos permiten en nuestro caso ampliar nuestro dataset generando una serie de imágenes, provocando así una explosión en el número de muestras que facilitan el entrenamiento de una red con un dataset de pocos recursos.

Además según [33] y cito

”Hemos propuesto un marco de aprendizaje conjunto para que las parejas de extremo a extremo vuelvan a aprender y a generar imágenes en una red unificada. Existe un bucle interactivo en línea entre los módulos discriminatorio y generativo para beneficiar mutuamente a ambas tareas. Nuestros dos módulos están co-diseñados para permitir que el aprendizaje de re-id aproveche mejor los datos generados, en lugar de simplemente entrenarse en ellos. Los experimentos en tres puntos de referencia demuestran que nuestro enfoque aporta consistentemente mejoras sustanciales tanto en la calidad de la generación de imágenes como en la precisión de la re-id.”

cosa que aprovechamos en nuestro pipeline.



**Figura 4.1:** Ejemplos de diferentes generaciones de datos sintéticos con redes generativas.

3. **Python.** Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en la legibilidad de su código. Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, dinámico y multiplataforma. El proyecto se encuentra programado en su totalidad en este lenguaje.
4. **OpenCv.** OpenCV es una biblioteca libre de visión artificial originalmente desarrollada por Intel. OpenCV significa Open Computer Vision (Visión Artificial Abierta). Desde que apareció su primera versión alfa en el mes de enero de 1999, se ha utilizado en una gran cantidad de aplicaciones, y hasta 2020 se la sigue mencionando como la biblioteca más popular de visión artificial. Detección de movimiento, reconocimiento de objetos, reconstrucción 3D a partir de imágenes, son sólo algunos ejemplos de aplicaciones de OpenCV.

Esta herramienta nos ha permitido realizar toda la gestión y procesamiento de imágenes y vídeo.

#### 5. **Pytorch.**

PyTorch es una biblioteca de aprendizaje automático de código abierto basada en la biblioteca de Torch, utilizado para aplicaciones que implementan cosas como visión artificial y procesamiento de lenguajes naturales, principalmente desarrollado por el Laboratorio de Investigación de Inteligencia Artificial de Facebook (FAIR). Es un software libre y de código abierto liberado bajo la Licencia Modificada de BSD. A pesar de que la interfaz de Python está más pulida y es el foco principal del desarrollo, PyTorch también tiene una interfaz en C++.

PyTorch proporciona dos características de alto nivel:

- Computación de tensores (como NumPy) con una aceleración fuerte a través de unidades de procesamiento gráfico (GPU).
- Redes neuronales profundas construidas en un sistema de Diferenciación Automática de Bases de Datos.

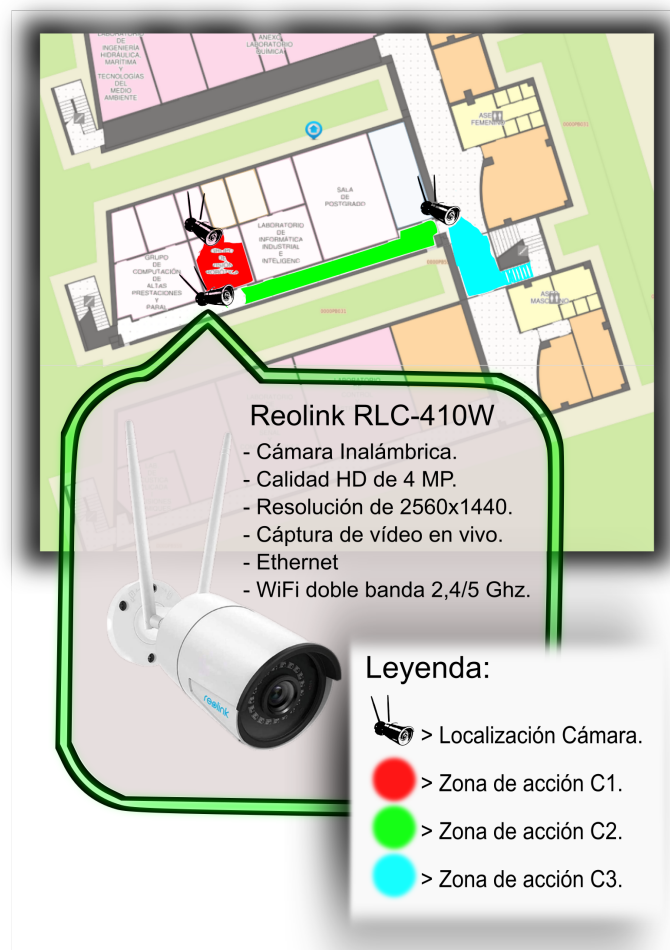
#### 6. **CUDA.**

CUDA son las siglas de Compute Unified Device Architecture (Arquitectura Unificada de Dispositivos de Cómputo) que hace referencia a una plataforma de computación en paralelo incluyendo un compilador y un conjunto de herramientas de desarrollo creadas por NVidia que permiten a los programadores usar una variación del lenguaje de programación C para codificar algoritmos en GPU de nVidia.

---

#### 4.2.1 Sistema de cámaras, hardware y descripción del entorno.

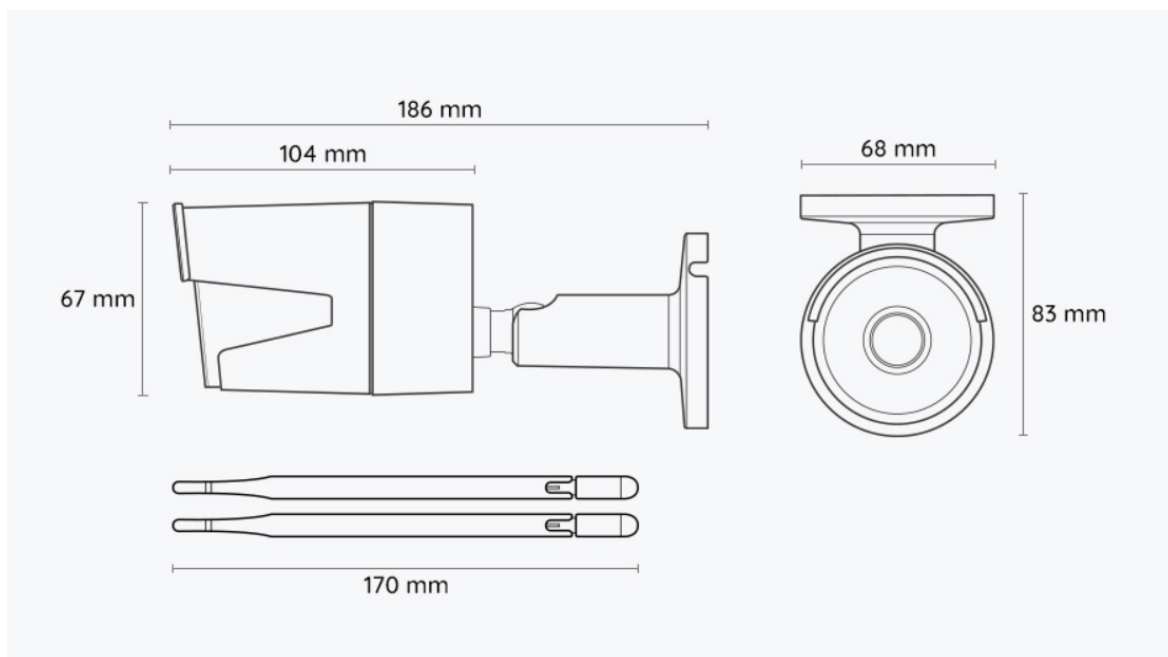
El entorno de pruebas consta de 3 cámaras en diferentes localizaciones de la Escuela Politécnica III (EPS III), dispuestas de manera que la oclusión entre las diferentes zonas de acción entre ellas es despreciable, con lo cual el entorno multicámara no presenta puntos en común entre las diferentes cámaras. Las cámaras son todas iguales, se trata del modelo Reolink RLC-410W. El entorno es abierto, pero hay secciones que pueden verse afectadas por la luminosidad, lo cual puede afectar a la reidentificación.



**Figura 4.2:** Diagrama cámaras y zonas.

#### 4.2.1.1 La cámara Reolink RLC-410W:

Lás cámaras presentan unas dimensiones que se pueden ver en la Figura 4.3 y unas especificaciones que se pueden ver en la figura 4.4 , y se localizan en la la EPS II, concretamente ver Figura 4.2 y desde ellas se alcanzan a ver 3 salas, Figura 4.5.



**Figura 4.3:** Dimensiones facilitadas por el fabricante[22].

RLC-410W-4MP Cámara-IP-Especificaciones		
	Modelo	RLC-410W(Versión de 4MP)
Cámara	Sensor de Imagen	1/2.7" CMOS Sensor
	Píxeles Efectivos	2560x1440 (4.0 Megapíxeles)
	Objetivo	f=4.0mm F=2.0
	Ángulo de Visión	Horizontal: 80°, Vertical: 42°
	Modo día / noche	Cambio Automático
	Iluminación Mínima	0 Lux (Con Iluminador IR)
	Distancia IR	30 Metros (LED: 18/14mil/850nm)
	Compensación de Luz de Fondo	Admite
Video	Reducción de Ruido	3D DNR
	Compresión	H.264
	Resolución	Main Stream: 2560x1440, 2048x1536, 2304x1296; Sub Stream: 640x480
	Velocidad de Fotogramas Máxima	25fps
Audio	Interfaz	Con Micrófono Incorporado
Red	Interfaz	1 * 10M/100Mbps RJ45
	Protocolo de Red	HTTPS, SSL, TCP/IP, UDP, UPNP, RTSP, SMTP, NTP, DHCP, DNS, DDNS, FTP, P2P
	Navegadores Compatibles	IE, Edge, Chrome, Firefox, Safari
	OS Compatibles	PC: Windows, Mac OS; Teléfono Inteligente: iOS, Android
	Acceso máx. de usuario	20 usuarios (1 cuenta de administrador y 19 cuentas de usuario); Admite hasta 12 video streams simultáneos (10 substreams & 2 mainstreams).
WiFi	Almacenamiento	Ranura para Tarjeta Micro SD, admite grabación de detección de movimiento
	Estándar Inalámbrico	IEEE 802.11a/b/g/n
	Frecuencia de Funcionamiento	2.4GHz / 5GHz
General	Seguridad Inalámbrica	WPA-PSK/WPA2-PSK
	Fuente de Alimentación	DC12V
	Consumo de Energía	<8W
	Ambiente de Trabajo	-10°C~+55°C (14°F~131°F), 10%~90%
	Grados de Protección	IP66
	Dimensiones	Φ67 x 186mm
	Peso	425g

**Figura 4.4:** Tabla de especificaciones de la cámara usada en el sistema [22].

#### 4.2.1.2 El entorno de pruebas:

El entorno de pruebas cubre tres áreas de la EPS III de la Universidad de Alicante, estas estancias son contiguas y consisten en:

1. La C1, una sala pequeña de despacho denotada en color rojo en la Figuras 4.5 y 4.2.
2. La C2, una pasillo con vidriera que sale de la C1 hacia la C3. Este pasillo comunica con otros despachos y salas ajenos al sistema. Se trata de zonas de oclusión fuera de nuestro sistema. Esta zona se marca en color verde en la Figuras 4.5 y 4.2, respectivamente.
3. La C3, un pasillo general del segundo piso de la EPS III que enfoca hacia parte de la escalera de acceso al segundo piso. Esta zona se marca en color azul en la Figuras 4.5 y 4.2 respectivamente.

C1- Zona cámara 1.



C2- Zona cámara 2.



C3- Zona cámara 3.



Figura 4.5: Las zonas a vista de cámara.

#### 4.2.2 Ordenadores utilizados en el desarrollo

Durante el proceso de desarrollo se han empleado diferentes máquinas. Cabe destacar que se ha empleado un servidor para poder ejecutar los scripts que requieren de CUDA[7], mientras que el desarrollo de los scripts se ha realizado en un portátil (listado al final).

- **Servidor - ServerGPU**

**SO:** Ubuntu 18.04.5

**GPU 1:** GeForce RTX 2080

**GPU 2:** GeForce RTX 2080

**GPU 3:** GeForce GTX 1080

**CUDA:** version 11.1

- **Portátil**

**SO:** Ubuntu 16.04/ Windows 10

**CPU:** AMD Ryzen 7 3750H APU

**RAM:** DDR IV 8GB\*2 (2666MHz)

**Tarjeta gráfica:** AMD Radeon RX 5500M 4GB GDDR6

**CUDA:** No compatible.

---

## 4.3 El dataset market-1501

### 4.3.1 Características generales

El dataset Market-1501 es un dataset para reidentificación de personas. Se recolectó en frente de un supermercado en la universidad de Tsinghua. Por lo tanto, este dataset se ha recogido en un entorno abierto, concretamente se recogió en verano de 2015.

Para su recolección se emplearon un total de 6 cámaras, incluyendo entre ellas 5 cámaras de alta resolución y una de baja resolución. Entre estas cámaras existe solapamiento.

En resumen, este dataset contiene 32,668 bounding boxes pertenecientes a 1501 identidades, cada una de las cuales está presente en al menos dos cámaras del sistema. Por lo tanto, se puede hacer una reidentificación a partir de cualquiera de esas 1501 clases o individuos.

### 4.3.2 Organización de los ficheros.

Su contenido (.zip) consta de la siguiente estructura:

- **Carpeta bounding box test** : Esta contiene 19732 imágenes para test.
- **Carpeta bounding box train** : contiene 12936 imágenes para entrenamiento.
- **Carpeta query** : contiene 3368 imágenes query. La búsqueda se realiza en la **carpeta bounding box test**.
- **Carpeta gt bbox** : contiene 25259 imágenes, todas ellas categorizadas a mano. Estas imágenes se corresponden con los 1501 individuos de los conjuntos de test y entrenamiento. Es usado para discriminar entre bueno, malo y basura y distracciones.
- **Carpeta gt query** : Por cada 3368 queries o consultas, hay tantas imágenes buenas como imágenes basura relevantes (que contiene el mismo individuo). Este archivo contiene un índice de las imágenes buenas y basura, este índice se emplea en la evaluación de desempeño.

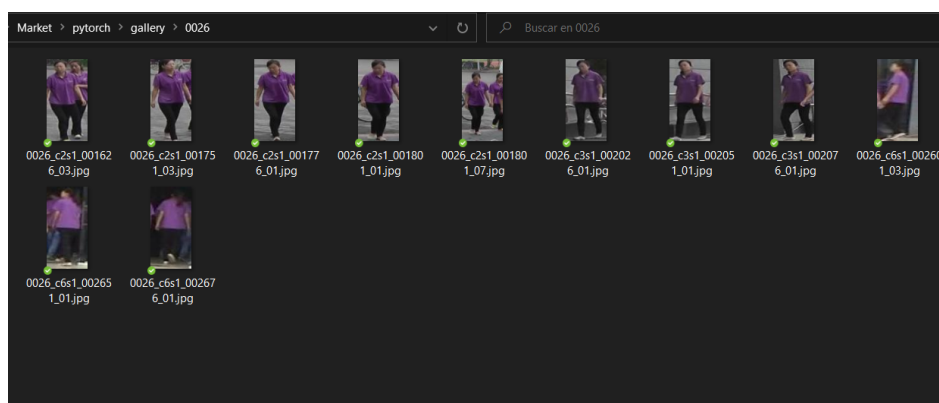


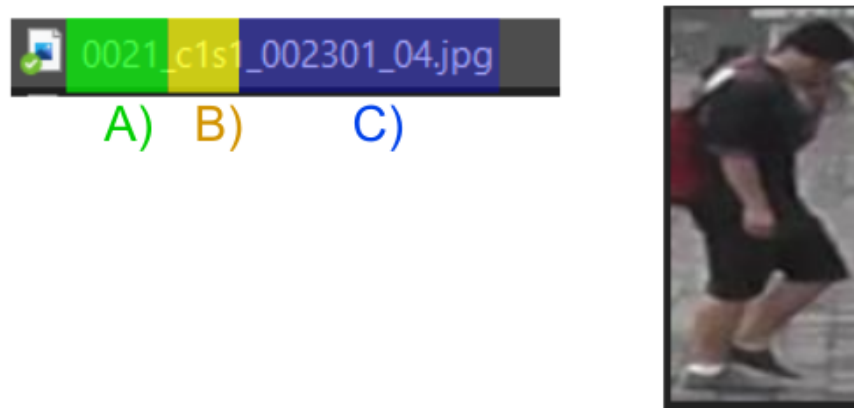
Figura 4.6: Contenido de Market-1501 individuo 0026.



### 4.3.3 Características de las imágenes

Estas imágenes tienen una serie de características:

- Tamaño de 64 x 128 píxeles.
- Cada una de estas imágenes sigue una estructura en su nombre, ver figura 4.7.



**Figura 4.7:** El *name rule* obedece lo siguiente:

- A: Se corresponde con la identificación del individuo dentro del dataset, de 0000 a 1501.
- B: Se corresponde con los datos de la cámara (de 1 a 6 en el dataset)
- C: Información extra no relevante para el pipeline. .

*nota:* se ha mantenido esta **name rule** para el desarrollo de nuestro proyecto, simplemente se hace una traducción de la etiqueta *id* para nuestros nuevos **targets**, añadiendo individuos a partir del 1501

## 4.4 Instalación y entorno virtual en CONDA, prerequisites y librerías.

En este anexo se va a documentar el proceso necesario para preparar un entorno e instalar todos los bloques funcionales del proyecto, librerías, puesta en funcionamiento de Yolov3[21], pytorch[17] etc.

### 4.4.1 Instalación de dependencias Reidentificación de personas.

A continuación listamos los prerequisites para el bloque de Re-ID:

- Python 3.6

- GPU Memory  $\geq$  6G
- Numpy
- Pytorch 0.3+
- apex (for float16) [Optional]
- pretrainedmodels [Optional]

**nota:** es necesario un sistema con CUDA [7], así como una GPU con buenas prestaciones para la correcta ejecución del sistema.

De cara a instalar todos los paquetes necesarios para poder ejecutar el sistema, es recomendable utilizar un entorno virtual de Python para mantener la versión del sistema operativo intacta. Durante el desarrollo de este proyecto se ha utilizado un entorno virtual de Python 3.6 gestionado mediante Anaconda.

Los pasos para instalar Anaconda en Ubuntu 16.04 han sido los siguientes:

1. Descargar el instalador de Anaconda para Ubuntu desde su web oficial.
2. Copiar el archivo al directorio home y ejecutar el siguiente comando:
3. Seguir las instrucciones del instalador y al acabar activar la instalación:

Una vez instalado Anaconda, se han utilizado los siguientes comandos para crear y utilizar un entorno virtual:

1. Crear el entorno virtual, en nuestro caso llamado reID:

```
1 conda update conda
2 conda create -n reID python =3.6 anaconda
```

2. Instalar paquetes en el nuevo entorno virtual:

```
1 conda update conda
2 conda create -n reID python =3.6 anaconda
```

3. Activar el entorno cada vez que lo vayamos a utilizar:

```
1 conda activate reID
```

4. Desactivar el entorno cuando lo dejamos de utilizar:

```
1      conda deactivate
```

Cuando tenemos listo nuestro entorno virtual, podemos pasar a instalar las dependencias concretas.

- Instalar pytorch de <http://pytorch.org/>
- Instalar pretrainedmodels

```
1      pip install pretrainedmodels
```

Ahora los prerequisites para el bloque de detección de personas mediante YOLOv3:

- Instalar darknet para yoloV3

```
1      git clone \url{https://github.com/pjreddie/darknet}
2      cd darknet
3      make
```

- Instalar OpenCv.
-

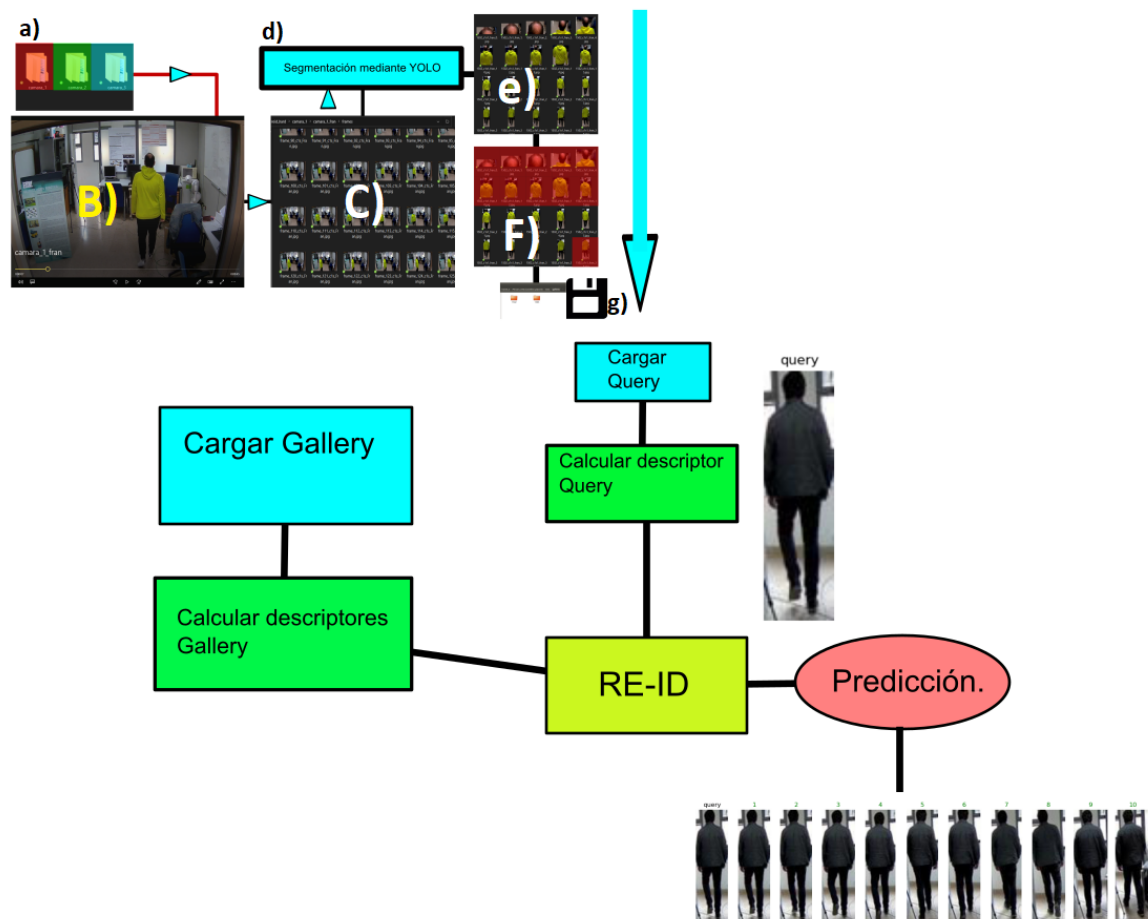


## 5 Desarrollo.

En este capítulo se va a explicar cómo se han llevado a cabo los diferentes pasos para completar el proyecto y se expondrán las diferentes decisiones que se han tomado para llevar el proyecto a su estado final.

### 5.1 Pipeline del sistema de reidentificación de personas.

El flujo de trabajo en alto nivel obedece al siguiente esquema:

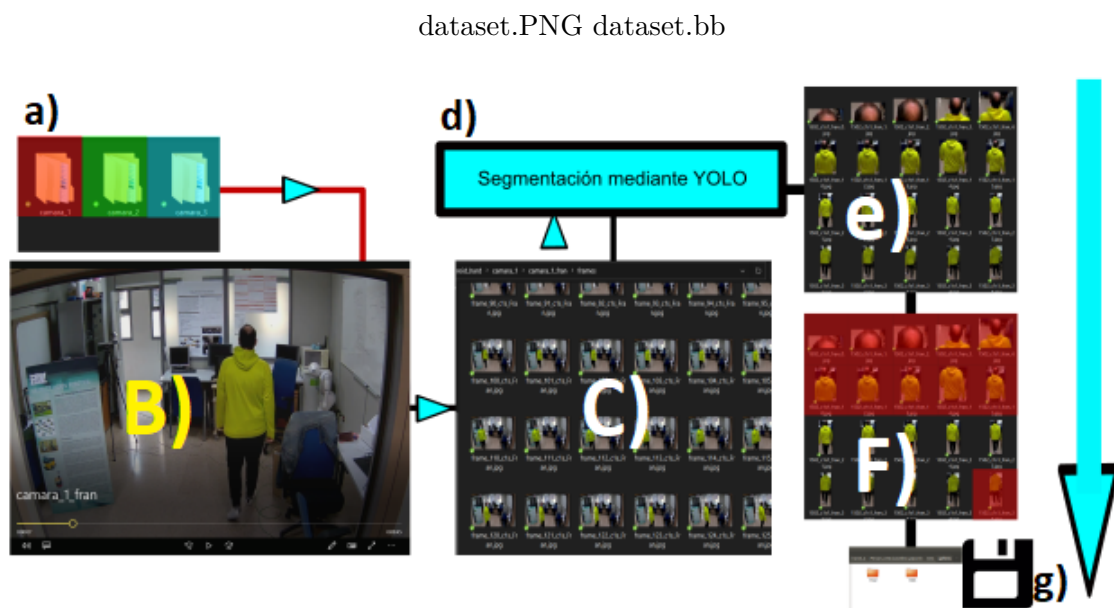


**Figura 5.1:** Workflow de alto nivel de nuestra aproximación a Re-ID.

## 5.2 Dataset

Dado que necesitamos realizar una Re-ID de unos targets (personas a Re-ID), es necesario generar nuestro propio dataset de targets.

Para la experimentación se han tomado vídeos por separado de estos targets con objeto de obtener los frames más prometedores y extraer de los mismos imágenes viables para luego realizar una Re-ID favorable. Nuestro dataset hereda la organización del dataset Market-1501 explicado anteriormente en el apartado de Metodología, dataset que se empleará para entrenar la red que realizará la Re-ID. Siguiendo por tanto la misma regla de nomenclatura. Para el experimento se han propuesto dos targets, Fran y Miguel. A nivel de código se han llamado 1502 y 1503 respectivamente, estos códigos luego se traducen para mostrarse en pantalla.



**Figura 5.2:** Workflow de la segmentación del target 1502 (Fran) en la sección 1 del entorno.

Los vídeos generados a partir de la cámaras son lo siguientes:

- **C1 - Cámara 1:**

*camara\_1\_fran.mp4.*

*camara\_1\_miguel.mp4.*

*camara\_1\_test.mp4.*

- **C2 - Cámara 2:**

*cámara\_2\_fran.mp4.*

*cámara\_2\_miguel.mp4.*

*cámara\_2\_test.mp4.*

- **C2 - Cámara 3:**

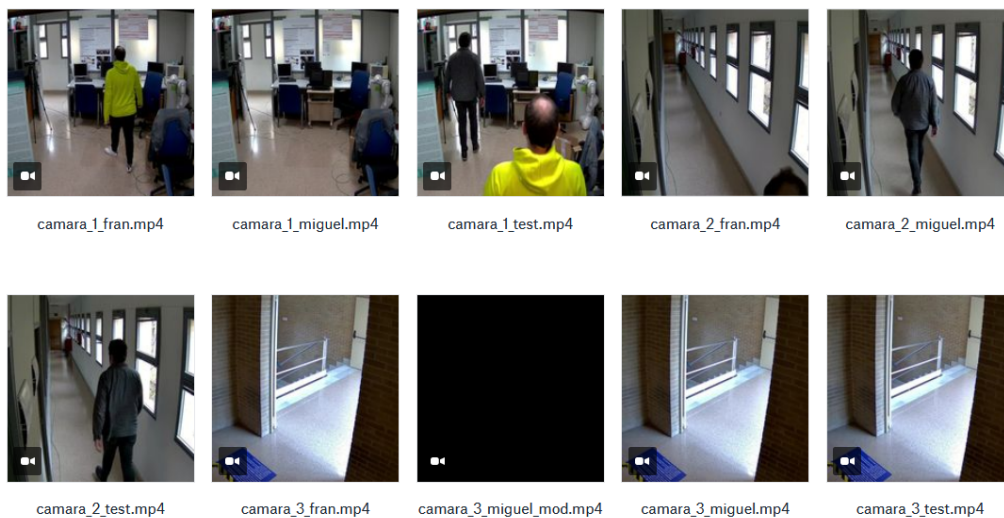
*cámara\_3\_fran.mp4.*

*cámara\_3\_miguel.mp4.*

*cámara\_3\_test.mp4.*

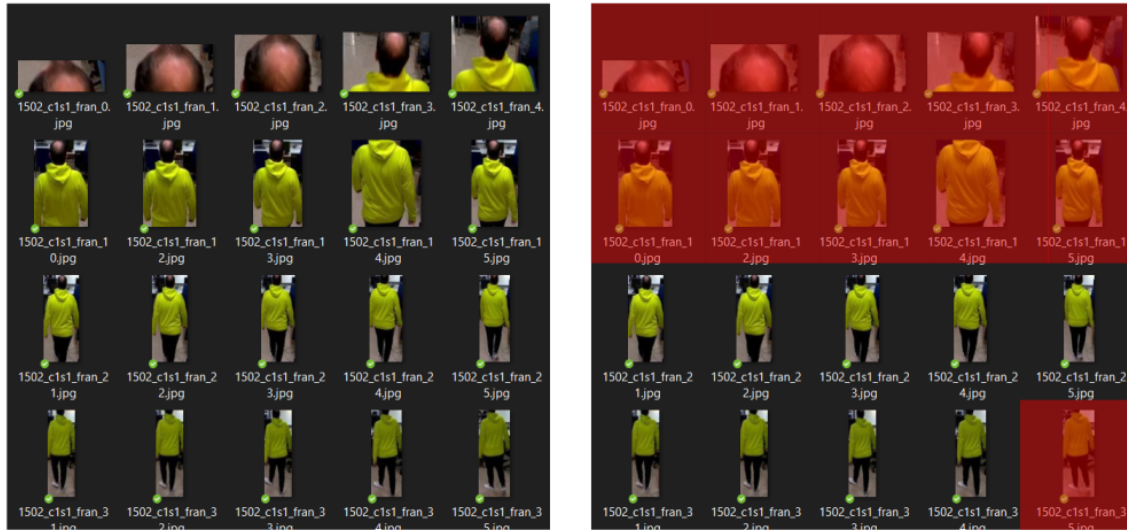
reid\_hard

Ordenado por nombre



**Figura 5.3:** Los diferentes vídeos generados por nuestro sistema de cámaras.

Una vez recogidos los vídeos el dataset se pasa por una segmentación mediante Yolo [21] donde frame a frame reconocemos los bounding box del target, para posteriormente realizar un crop de dicho frame y guardarlo en la carpeta persons, correctamente etiquetados (recordemos que en los vídeos para segmentación solo sale nuestro target, con lo que el etiquetado es sencillo). Tras esto se realiza un cribado manual de aquellos frames no prometedores como se ve en la siguiente figura 5.4.



**Figura 5.4:** El proceso de cribado de la segmentación con yolo de los targets se hace a mano, eliminando aquellas imágenes no prometedoras.

## 5.3 Realización de la reID

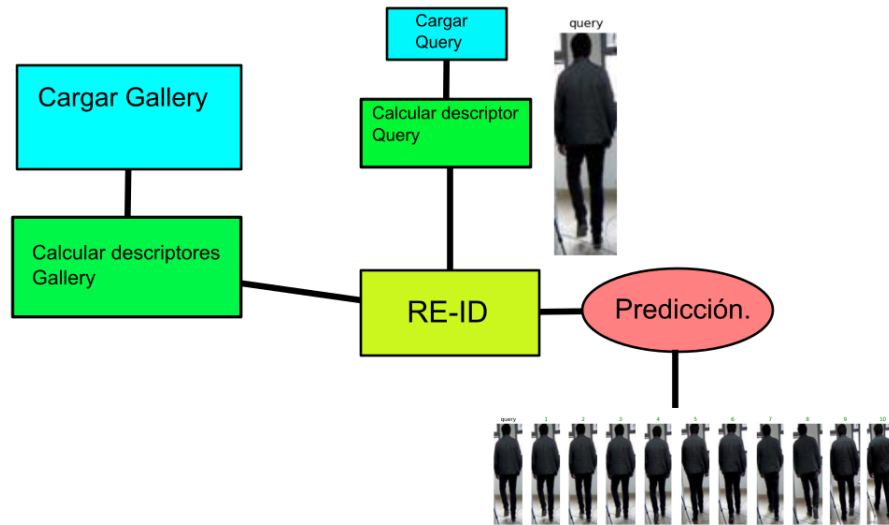
### 5.3.1 Re-ID por imágenes del dataset

Una vez tenemos el dataset de nuestros targets preparado, podemos pasar a abordar el problema de la reidentificación. Partimos de realizar unas pruebas fuera de tiempo real. Seleccionando una imagen target y buscando en toda la galería con nuestra red entrenada en pytorch. La red nos muestra las 10 imágenes más prometedoras (en términos de la red viene a ser aquellas imágenes cuyos descriptores que guardan menos distancia con el descriptor de nuestro query). Para ver la secuencia de pasos, ver figura 5.5.

### 5.3.2 Re-ID en tiempo real

Una vez solucionado el problema anterior, aplicamos dicho procesamiento en tiempo real. Se trata de ir un paso más allá, pasamos a segmentar nuestro query de cada frame de un vídeo, y pasárselo a la red para realizar el cribado de ese bounding box y devolver la clase a la que pertenece nuestro target, es decir, a qué clase pertenece la imagen de nuestro dataset cuyo descriptor guarda menor distancia con nuestro query. Y así con todos los targets detectados en el frame por yolo, ver Figura 5.6.





**Figura 5.5:** Workflow de la reidentificación del sujeto 1503 (Miguel).

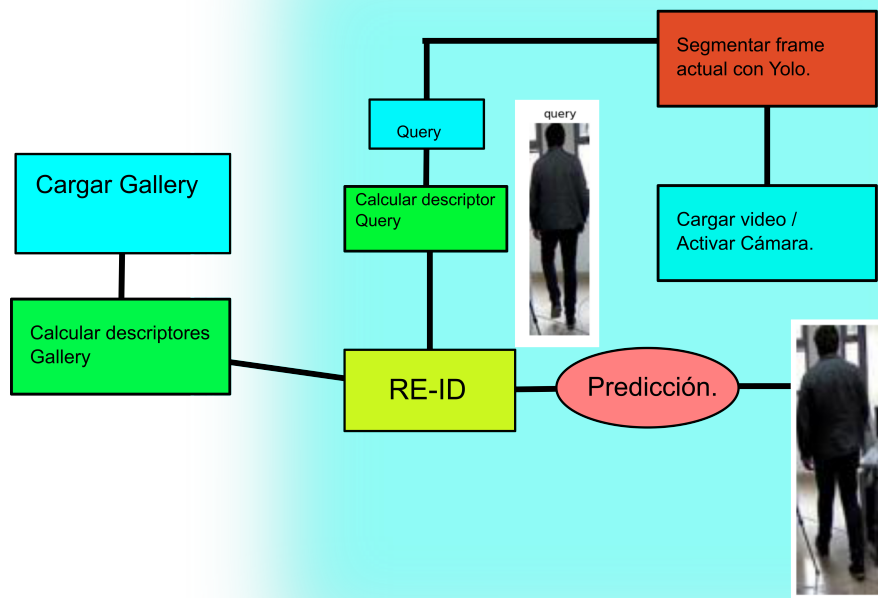
## 5.4 El entrenamiento de la red con Market-1501

A pesar de haber optado por usar un modelo preentrenado con el dataset Market-1501, podemos realizar un entrenamiento de la red, introduciendo nuestro dataset de targets. Para ello, añadimos las carpetas de nuestros targets, al dataset market, al haber respetado la name rule, con nuestro nuevo dataset, simplemente tenemos que añadir al sistema de carpetas, las carpetas 1502 y 1503. Y pasaríamos a realizar el entrenamiento.

### 5.4.1 Entrenar bajo la propuesta de mixed training

Para entrenar nuestro sistema se sirve de una combinación de generación de imágenes sintéticas mediante el uso de redes GAN de manera simultánea al entrenamiento [34]. Una visión general esquemática de la DG-Net se puede ver en la figura 5.7. (a) Nuestro módulo de aprendizaje discriminatorio de reid está incrustado en el módulo generativo compartiendo el codificador de apariencia  $E_a$ . Una línea negra de guión denota que la imagen de entrada del codificador de estructura  $E$  se convierte en gris. La línea roja indica que las imágenes generadas se retroalimentan en línea a  $E_a$ . En el módulo generativo se imponen dos objetivos: b) La generación de identidades propias mediante la misma identidad de entrada y c) La generación de identidades cruzadas mediante diferentes identidades de entrada. d) Para aprovechar mejor los datos generados, el aprendizaje de reid implica el aprendizaje de características primarias y la minería de características de grano fino.

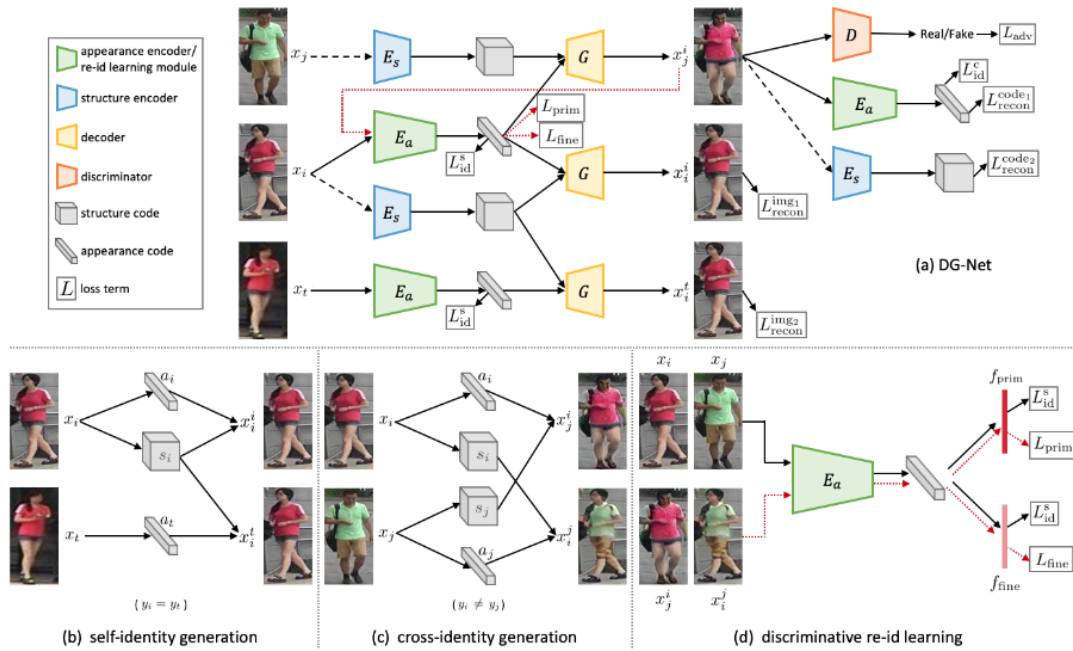
”En este trabajo proponen un marco de aprendizaje conjunto donde unifican en una red las parejas de extremo a extremo, el aprendizaje reid y la generación de imágenes. Existe un bucle interactivo en línea entre los módulos discriminatorio y generativo para beneficiar mutuamente a ambas tareas. Nuestros dos módulos están co-diseñados para permitir que el aprendizaje de re-id aproveche mejor los



**Figura 5.6:** Workflow de la reidentificación del sujeto 1503 (Miguel) en tiempo real.

datos generados, en lugar de simplemente entrenarse en ellos. Los experimentos en tres puntos de referencia demuestran que nuestro enfoque aporta consistentemente mejoras sustanciales tanto en la calidad de la generación de imágenes como en la precisión de la re-id.”

Esta aproximación mejora el entrenamiento, al coordinar la generación de nuevos datos sintéticos con el entrenamiento.



**Figura 5.7:** Visión general esquématica de la DG-Net, mixed train, con red GAN.



## 6 Resultados

Pasamos a discutir diferentes ejecuciones del pipeline de Re-ID. A continuación se muestran diferentes figuras resultado de la experimentación. En cada figura se muestra el query y las 10 imágenes más parecidas de la galería de targets (aquellas imágenes cuya distancia entre descriptores sea menor respecto al descriptor del query).

Pasamos a comentar cada una de estas figuras:

1. **Figura 6.1 :** En esta entrada de experimentación se ha cargado una imagen query de nuestro dataset de targets sin incluirlo en la galería de Market-1501, debido a eso, no se realiza una buena predicción del mismo, colecta imágenes principalmente de la carpeta -1 (una carpeta llena de imágenes de no targets o partes de targets sin identificar).
2. **Figura 6.2 :** Esta figura es interesante, se selecciona una imagen del sujeto 1502 (Fran) con una oclusión parcial y, sin embargo, el pipeline es capaz de encontrar imágenes para identificarlo de forma correcta.
3. **Figura 6.3 :** En esta ocasión se ha probado a introducir sólo uno de nuestros targets a la galería. Mientras que el otro se presenta como query, el pipeline establece que el candidato más óptimo es una imagen de target introducido en la galería. Cosa que era de esperar. Sin embargo, esta eventualidad se controla con umbrales pues la distancia entre descriptores es mucho más grande de lo esperado. Esto es lo que ocurriría si una persona externa al experimento entrara en la zona. Al no ser parte del target la predicción sería errónea, pero mediante un umbral en la distancia se desestimaría como No target.
4. **Figura 6.4 :** Esta figura muestra una oclusión parcial sólo se ve el torso de nuestro target 1502, de nuevo se realiza la reidentificación de forma exitosa.
5. **Figura 6.5 :** En esta figura se muestra una Re-ID exitosa del target 1503.
6. **Figura 6.6 :** En esta figura se muestra una Re-ID exitosa del target 1502, en el entorno C2, con condiciones lumínicas desfavorables.
7. **Figura 6.7 :** En esta figura se muestra otra Re-ID exitosa del target 1503.
8. **Figura 6.8 :** En esta figura se muestra una Re-ID exitosa del target 1502, este resultado es interesante dado que propone diferentes imágenes como candidatos en diferentes poses (corresponde a una secuencia de vídeo en la que el target está girando sobre si mismo).
9. **Figura 6.9 :** En esta figura se intenta confundir a la red, el query es una imagen tomada por un dispositivo móvil de mayor resolución, tomada en otro día diferente y con otras condiciones lumínicas, y con ropa diferente (pero similar, de ser muy diferente la red lo trataría como No target), pese a ello la red, hace una correcta identificación.



**Figura 6.1:** Ejecución del pipeline para un query sin imágenes del mismo en el Market-1501.



**Figura 6.2:** Oclusión parcial de un query del target 1502, con un Re-id exitoso.



**Figura 6.3:** Un query del sujeto 1503, sin imágenes del mismo en la galería.



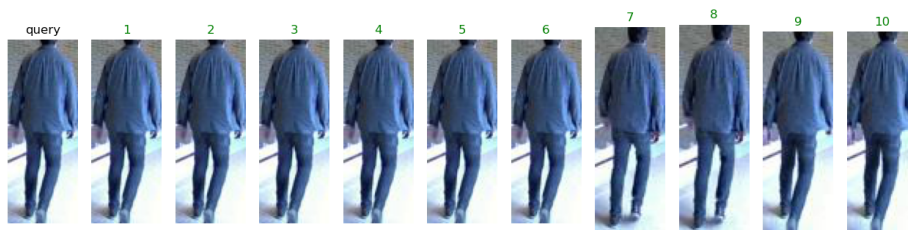
**Figura 6.4:** Ejemplo de ejecución para un query con oclusión parcial(sólo torso superior).



**Figura 6.5:** Correcta Re-Id del sujeto 1503.



**Figura 6.6:** Correcta reidentificación del sujeto 1502 en el entorno C2, con mala iluminación.



**Figura 6.7:** Ejemplo de re-id, del target 1503.



**Figura 6.8:** Detección del sujeto 1502, con diferentes poses como resultado.



**Figura 6.9:** Detección del sujeto 1503 tomada con un dispositivo móvil, en otra fecha, con ropa similar, y distinta iluminación.



## 7 Conclusiones

En este TFG hemos desarrollado un sistema completo que es capaz de solucionar el problema de la reidentificación. Nos hemos basado en un método basado en GAN para generar un descriptor de nuestro query (persona a Re-ID) que posteriormente se emplea para comparar dicho query con nuestra galería de targets (ver Figura 5.4) y poder realizar la predicción, en esencia se busca la imagen en nuestra galería de targets cuyo descriptor guarda menos distancia con el descriptor de nuestro target, y la clase de esa imagen de menor distancia se devuelve como nuestra predicción.

Se ha conseguido un sistema más o menos estable, fuera del loop de tiempo real, pero el salto a un sistema de tiempo real presenta una dificultad incremental que lleva a no ser viable mediante este método.

Una búsqueda estática bajo los términos de la red neuronal lleva a un buen desempeño, pero la brecha entre la experimentación en un laboratorio y la aplicación efectiva en un entorno real, es demasiado grande [2].

Hay variables críticas que no podemos salvar, ya sea cambios en la iluminación, errores en la predetección de los bounding boxes de nuestro query, o incluso la presencia de personas vestidas de forma similar. A mayor número de targets en la galería, más difícil será para nuestra red realizar la Re-ID de forma satisfactoria.

Por ello, concluyo que no es posible realizar el salto a la detección en tiempo real, al menos con esta aproximación. A día de hoy el problema de la re-id sigue siendo un problema sin resolver, salvo que se acoten las variables a controlar en el entorno.

Uno de los motivos que me llevan pensar que pueden influir negativamente en nuestro sistema es el tamaño de las imágenes, acotado por el dataset Marker-1501, al tener que reducir tanto las imágenes de nuestro target, se dificulta la re-Id. Sin embargo, no se están generando datasets de imágenes de gran tamaño. Esta es sin duda una de las opciones a explorar en proyectos futuros, testear el pipeline con una red entrenada con un dataset con imágenes de un tamaño mayor.



## Bibliografía

- [1] Jon Almazan, Bojana Gajic, Naila Murray, and Diane Larlus. Re-id done right: towards good practices for person re-identification. *arXiv preprint arXiv:1801.05339*, 2018.
- [2] Apurva Bedagkar-Gala and Shishir K Shah. A survey of approaches and trends in person re-identification. *Image and vision computing*, 32(4):270–286, 2014.
- [3] Gary Bradski and Adrian Kaehler. Opencv. *Dr. Dobb's journal of software tools*, 3, 2000.
- [4] Abir Das, Anirban Chakraborty, and Amit K Roy-Chowdhury. Consistent re-identification in a camera network. In *European conference on computer vision*, pages 330–345. Springer, 2014.
- [5] Abir Das, Rameswar Panda, and Amit Roy-Chowdhury. Active image pair selection for continuous person re-identification. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 4263–4267. IEEE, 2015.
- [6] Abir Das, Rameswar Panda, and Amit K Roy-Chowdhury. Continuous adaptation of multi-camera person identification models through sparse non-redundant representative selection. *Computer Vision and Image Understanding*, 156:66–78, 2017.
- [7] Rob Farber. *CUDA application design and development*. Elsevier, 2011.
- [8] Jorge Garcia, Niki Martinel, Alfredo Gardel, Ignacio Bravo, Gian Luca Foresti, and Christian Micheloni. Discriminant context information analysis for post-ranking person re-identification. *IEEE Transactions on Image Processing*, 26(4):1650–1665, 2017.
- [9] Niloofar Gheissari, Thomas B Sebastian, and Richard Hartley. Person reidentification using spatiotemporal appearance. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1528–1535. IEEE, 2006.
- [10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.
- [11] Douglas Gray and Hai Tao. Viewpoint invariant pedestrian recognition with an ensemble of localized features. In *European conference on computer vision*, pages 262–275. Springer, 2008.
- [12] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017.

- 
- [13] Ruibing Hou, Bingpeng Ma, Hong Chang, Xinqian Gu, Shiguang Shan, and Xilin Chen. Vrstc: Occlusion-free video person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7183–7192, 2019.
  - [14] Houjing Huang, Dangwei Li, Zhang Zhang, Xiaotang Chen, and Kaiqi Huang. Adversarially occluded samples for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5098–5107, 2018.
  - [15] Yukun Huang, Zheng-Jun Zha, Xueyang Fu, and Wei Zhang. Illumination-invariant person re-identification. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 365–373, 2019.
  - [16] Srikrishna Karanam, Yang Li, and Richard J Radke. Person re-identification with discriminatively trained viewpoint invariant dictionaries. In *Proceedings of the IEEE international conference on computer vision*, pages 4516–4524, 2015.
  - [17] Nikhil Ketkar. Introduction to pytorch. In *Deep learning with python*, pages 195–208. Springer, 2017.
  - [18] Martin Koestinger, Martin Hirzer, Paul Wohlhart, Peter M Roth, and Horst Bischof. Large scale metric learning from equivalence constraints. In *2012 IEEE conference on computer vision and pattern recognition*, pages 2288–2295. IEEE, 2012.
  - [19] Q. Leng, M. Ye, and Q. Tian. A survey of open-world person re-identification. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(4):1092–1108, 2020.
  - [20] Sébastien Marcel and Yann Rodriguez. Torchvision the machine-vision package of torch. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 1485–1488, 2010.
  - [21] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv*, 2018.
  - [22] Reolink. product rlc-410w specifications. <https://reolink.com/es/product/rlc-410w/#specifications>. Online; última visita 29 diciembre 2020.
  - [23] M Saquib Sarfraz, Arne Schumann, Andreas Eberle, and Rainer Stiefelhagen. A pose-sensitive embedding for person re-identification with expanded cross neighborhood re-ranking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 420–429, 2018.
  - [24] Jifei Song, Yongxin Yang, Yi-Zhe Song, Tao Xiang, and Timothy M Hospedales. Generalizable person re-identification by domain-invariant mapping network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 719–728, 2019.
  - [25] Yifan Sun, Liang Zheng, Weijian Deng, and Shengjin Wang. Svdnet for pedestrian retrieval. 2017.
  - [26] Taiqing Wang, Shaogang Gong, Xiatian Zhu, and Shengjin Wang. Person re-identification by video ranking. In *European conference on computer vision*, pages 688–703. Springer, 2014.
-

- 
- [27] Yan Wang, Lequn Wang, Yurong You, Xu Zou, Vincent Chen, Serena Li, Gao Huang, Bharath Hariharan, and Kilian Q Weinberger. Resource aware person re-identification across multiple resolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8042–8051, 2018.
  - [28] Ancong Wu, Wei-Shi Zheng, Hong-Xing Yu, Shaogang Gong, and Jianhuang Lai. Rgb-infrared cross-modality person re-identification. In *Proceedings of the IEEE international conference on computer vision*, pages 5380–5389, 2017.
  - [29] Qize Yang, Ancong Wu, and Wei-Shi Zheng. Person re-identification by contour sketch under moderate clothing change. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, page 1–1, 2020.
  - [30] Mang Ye, Chao Liang, Zheng Wang, Qingming Leng, Jun Chen, and Jun Liu. Specific person retrieval via incomplete text description. In *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, pages 547–550, 2015.
  - [31] Mang Ye, Jianbing Shen, Gaojie Lin, Tao Xiang, Ling Shao, and Steven C. H. Hoi. Deep learning for person re-identification: A survey and outlook, 2021.
  - [32] Wei-Shi Zheng, Shaogang Gong, and Tao Xiang. Towards open-world person re-identification by one-shot group-based verification. *IEEE transactions on pattern analysis and machine intelligence*, 38(3):591–606, 2015.
  - [33] Zhedong Zheng, Xiaodong Yang, Zhiding Yu, Liang Zheng, Yi Yang, and Jan Kautz. Joint discriminative and generative learning for person re-identification. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
  - [34] Zhedong Zheng, Liang Zheng, and Yi Yang. A discriminatively learned cnn embedding for person reidentification. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 14(1):13, 2018.
-



## Lista de Acrónimos y Abreviaturas

<b>EPS III</b>	Escuela Politécnica III.
<b>GAN</b>	<i>Generative Adversarial Networks.</i>
<b>IEEE</b>	<i>Institute of Electrical and Electronics Engineers.</i>
<b>Re-ID</b>	reidentificación.
<b>TFG</b>	Trabajo Final de Grado.